

Programare concurentă și distribuită

- MSD2 – Enache Rareș – Homework 1

1. Usage

To launch the server, one has to open a terminal (under a Linux machine) and run the following command:

```
python3 server.py <sock_proto> <buff_size>
```

→ where **<sock_proto>** is either *TCP* or *UDP*;

→ where **<buff_size>** is an integer value, representing the package size (defaults to 512 bytes);

To launch the client, one has to repeat the same steps, only replacing the file name (*server.py* → *client.py*) and to specify the filename to transfer as a third parameter. The file has to reside in the root directory of both *server.py* and *client.py*.

Due to file sizes, I did not upload any sample file. One will have to place some file under the root directory and specify its name as the third parameter when launching the client.

*Note: If the file name is not specified, it defaults to **send**.*

2. Testing

The system was tested using two files of different sizes. One was ~30MB, the other just above 1GB.

Over TCP/IP protocol, the 30MB file takes 0:00:00.126046 microseconds to be transferred from the client to the server. For the +1GB file, it takes 0:00:06.287847 seconds.

Over UDP protocol, the 30MB file takes just 0:00:01.615663 seconds to be transferred from the client to the server. For the +1GB file, it takes 0:00:58.383921 seconds. Also, around 400 packets were lost in the process for both files, due to the connectionless nature of the UDP protocol (no actual ACK of the packages received on the server side).

Note: Tests were done on localhost.

3. Statistics

	30MB	1GB
TCP	0,2022103 s	7,2831625 s
UDP	1,3795265 s	51,3609524 s

